# The Algorithmic Foundations of Data Privacy

Instructor: Aaron Roth

# Administrivia

http://www.cis.upenn.edu/~aaroth/courses/privacyF11.html

- Time: Tuesday/Thursday 1:30-3:00
- Room: Here (Towne 315)
- Format:
  - Lectures
  - Student Presentations of Projects
- Evaluation:
  - Class project (60%)
  - Participation (40%)
    - Including blog posts! http://privacyfoundations.wordpress.com/

# Administrivia

http://www.cis.upenn.edu/~aaroth/courses/privacyF11.html

- Project: Semester long study of a topic in privacy
  - Topic suggestions up soon on the website
    - Feel free to pick your own!
  - Can be pure theory, implementation, or somewhere in between
    - Literature review
    - Some component of original research
  - Graded components:
    - Proposal, mid project report, final report and presentation.

# Course Overview

- How can we perform private data analysis?
  - How do we mathematically define “Privacy”?
- How does “privacy” degrade when multiple analyses are performed?
- What are the theoretical limits of how much information we can release about a dataset while preserving “privacy”?

# Course Overview

- How can we design efficient algorithms that make use of data privately?
- How should economic agents reason about their privacy?
  - How should we design auctions and other mechanisms for privacy-aware consumers?

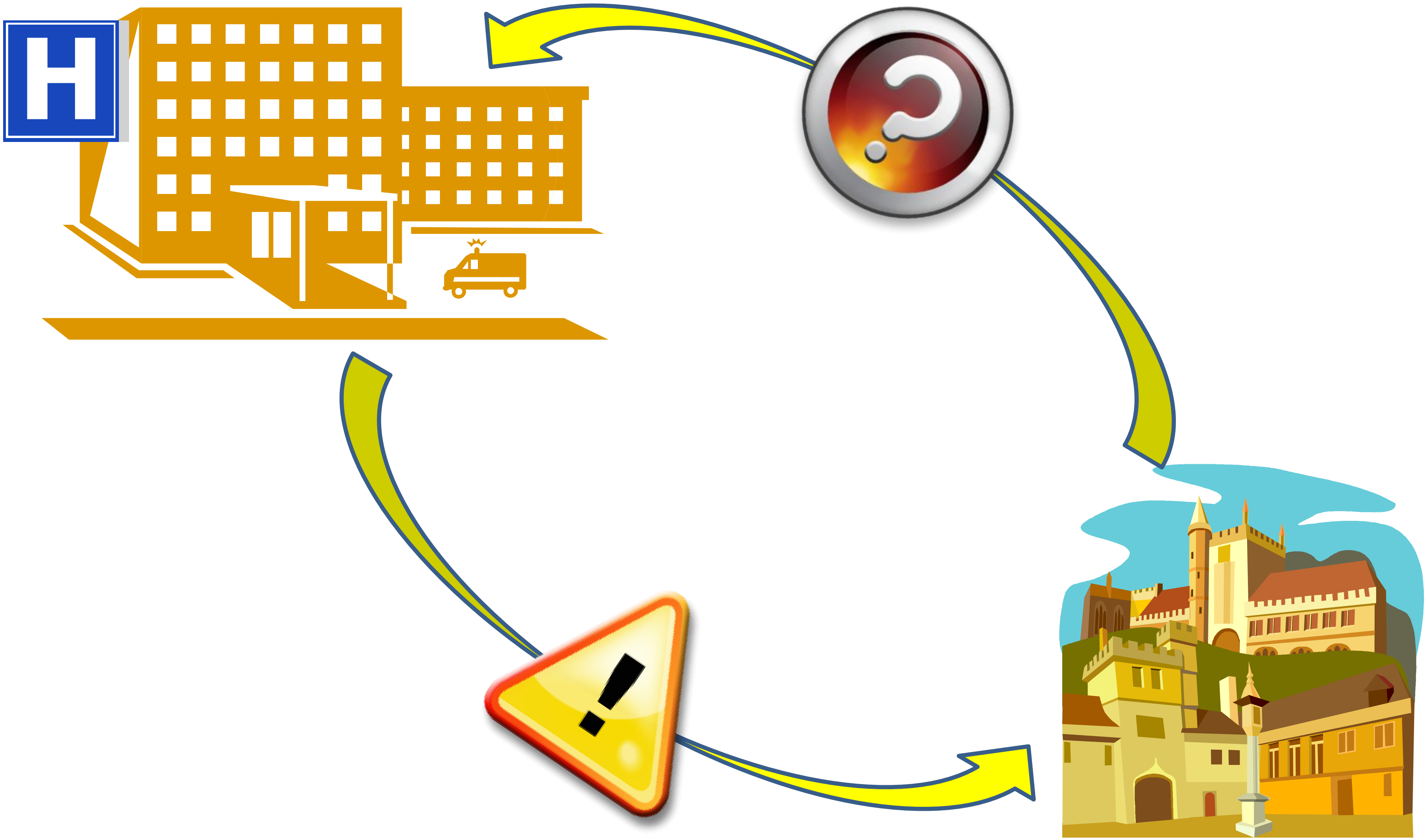
# Today

- Some motivation
- The definition of differential privacy
- An overview of topics we will cover
- If there is time: A lower bound.

# Warning

- Powerpoint: I will probably go too fast
- Stop me! Ask questions!
  - Other people probably have the same question.
  - I will be suspicious if you don’t…
  - Remember participation is 40% of your grade!

# A Dilemma

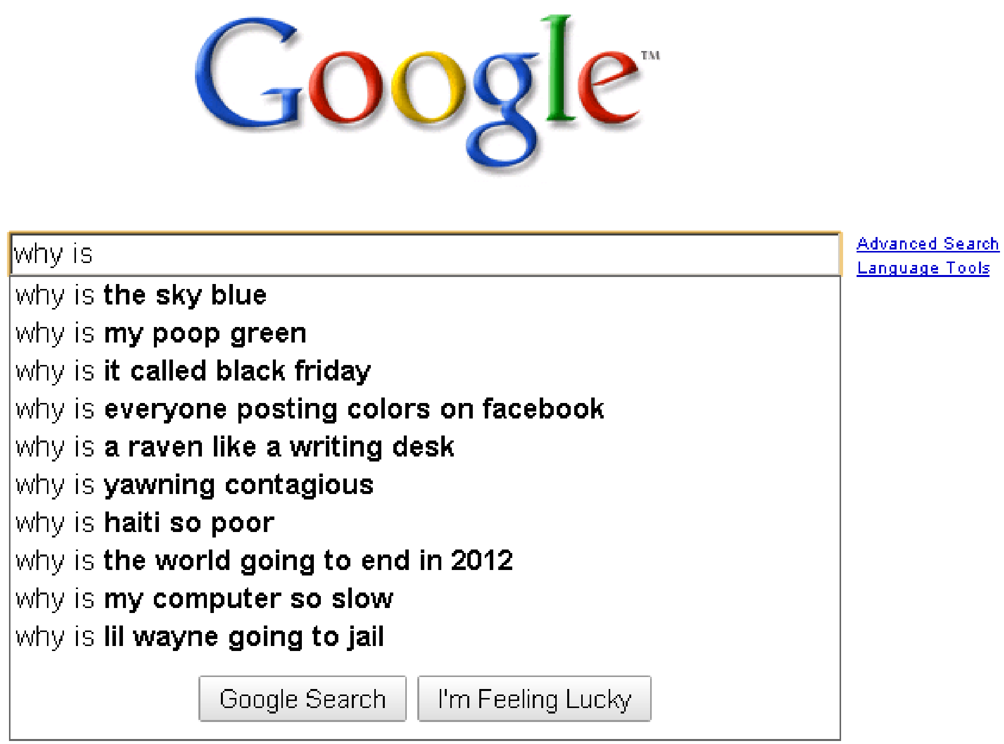


# Modern Algorithm Design

- Computation is not the only constraint
- Dealing with large datasets
  - Data *belongs* to other people
  - Must protect their privacy
  - Must convince them to report it truthfully

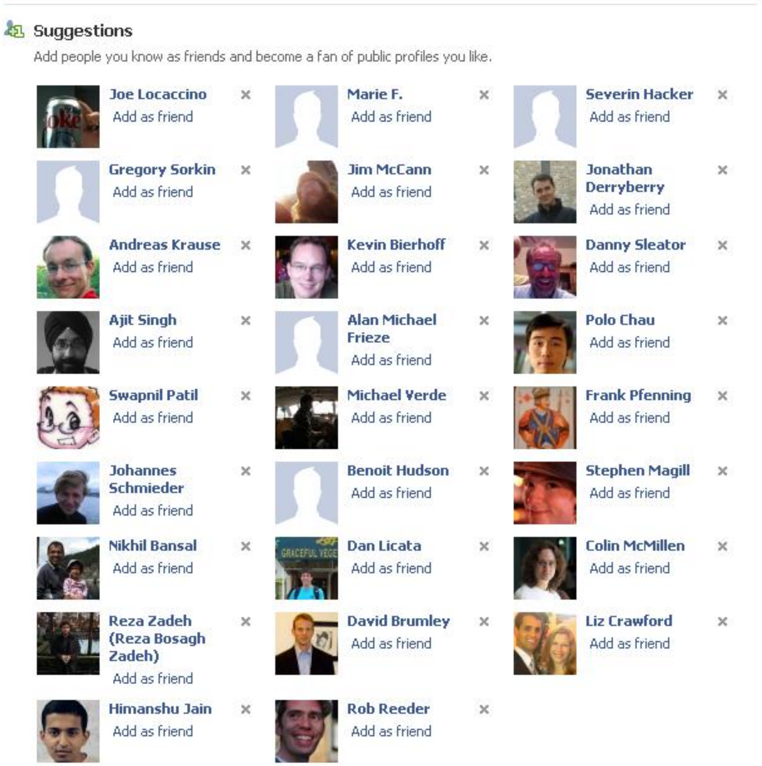
# Modern Algorithm Design

- Use search logs to recommend query completions



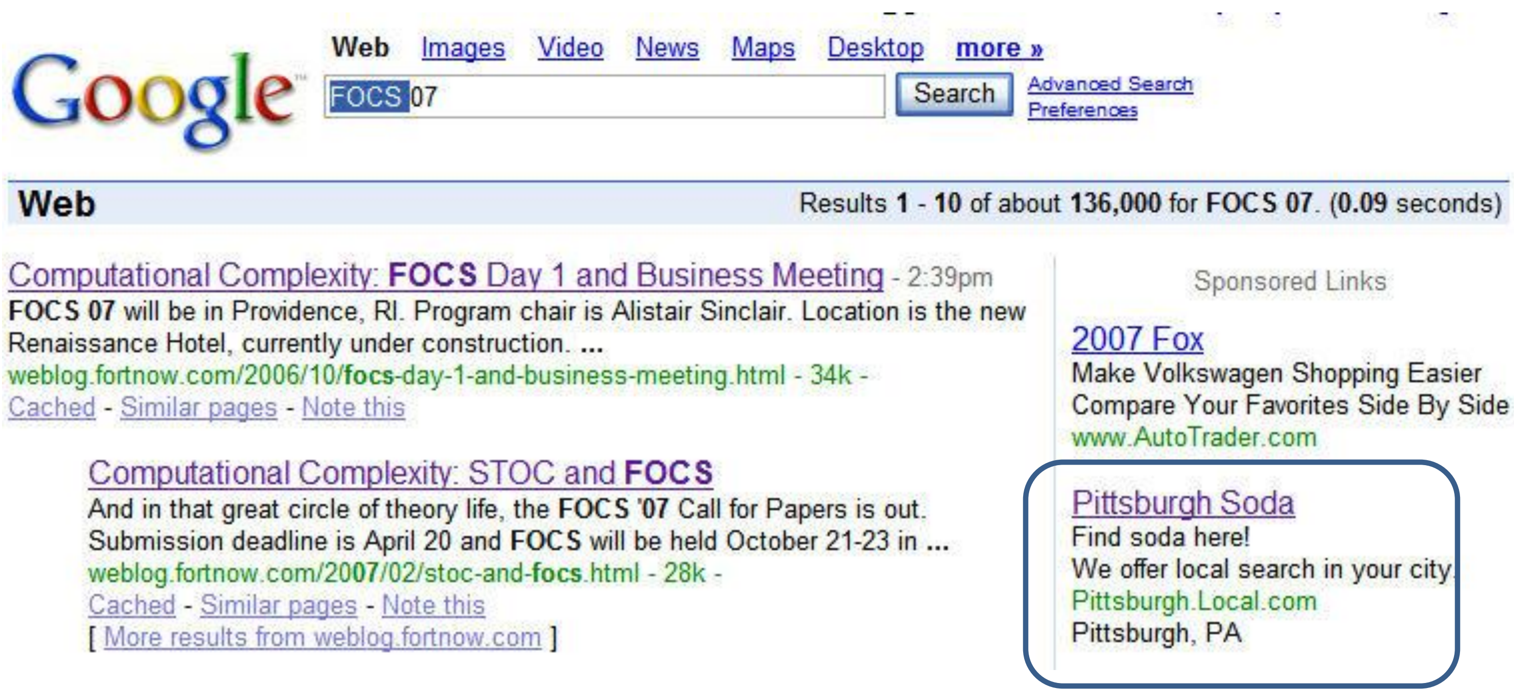
# Modern Algorithm Design

- Find closely connected components in a social network



# Modern Algorithm Design

- Decide which ads to show based on user data and other users previous searches.



# What is Privacy?

# What Isn’t Privacy?

- Privacy isn’t restricting questions to large populations.
  - “What is the average salary of Penn faculty?”
  - “What is the average salary of Penn faculty not named Aaron Roth?”

# What Isn’t Privacy?

- Privacy isn’t restricting to “ordinary” facts.
  - Statistics on Alice’s bread buying habits: For 20 years she regularly buys bread, and then stops.
    - Type 2 diabetes?

# What Isn’t Privacy?

- Privacy isn’t “Anonymization”
  - Anonymization is hard.
    - Problem: Auxiliary Information and Linkage Attacks!
    - Case Study: NetFlix Prize Dataset
      - Linked with IMDB database to re-identify users [Narayanan, Shmatikov]
      - 2nd Netflix prize cancelled
    - Can’t know what the adversary knows, or might know in the *future*.

# What Isn’t Privacy?

- Privacy isn’t “Anonymization”
  - Anonymization isn’t enough
    - Collection of medical records from a specific urgent care center and date might correspond to only a small collection of medical conditions.
    - Knowledge (from a neighbor?) that Alice went to that urgent care center doesn’t identify her record, but implies she has one of a small number of conditions.

# What is Privacy?

- Freedom from harm.

Privacy Definition, Attempt 1:

*An analysis of a dataset D is private if the data analyst knows no more about Alice after the analysis than he knew about Alice before the analysis.*

# What is Privacy

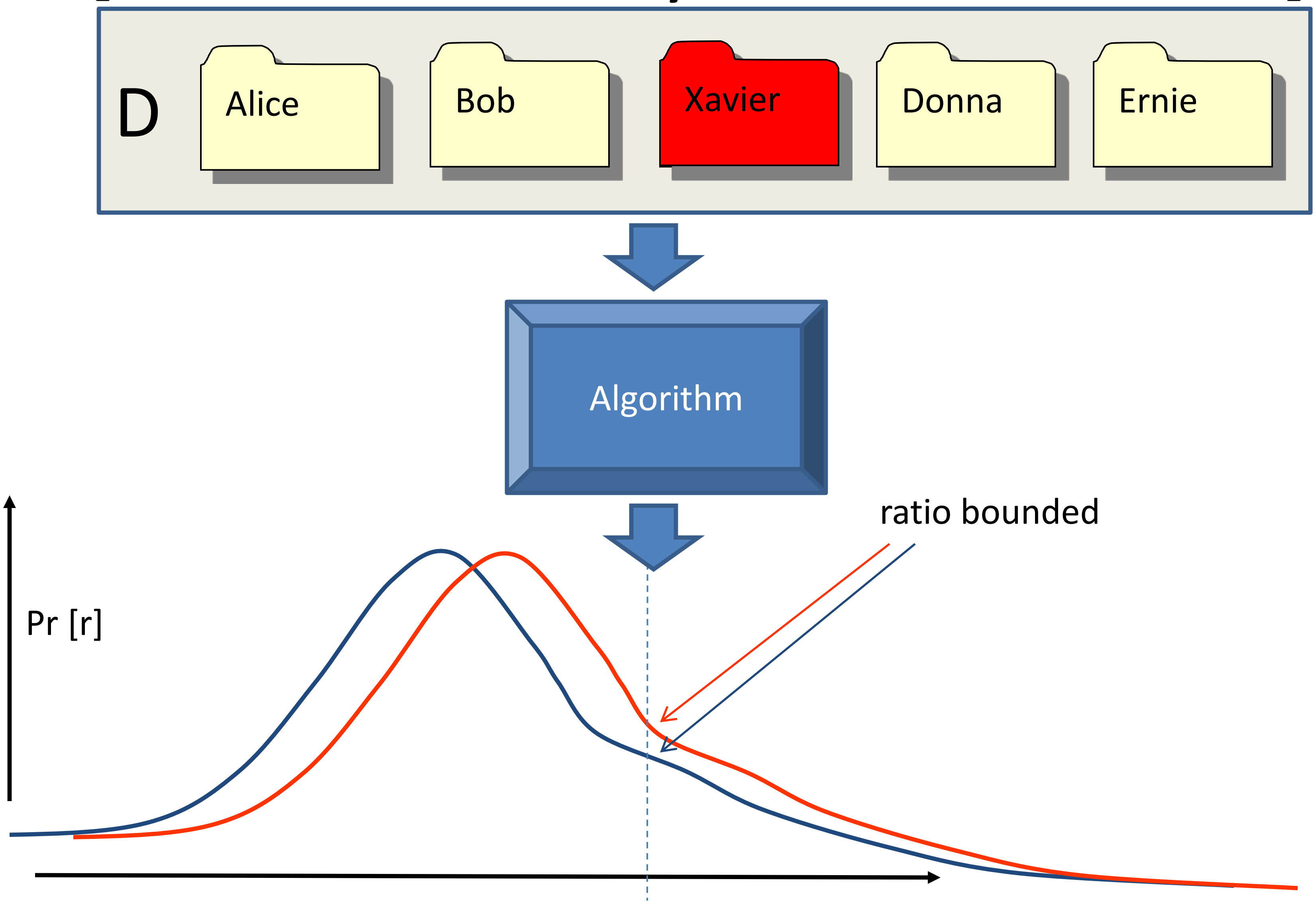
- Problem: Impossible to achieve with auxiliary information.
  - Suppose an insurance company knows that Alice is a smoker.
  - An analysis that reveals that smoking and lung cancer are correlated might cause them to raise her rates!
- Was her privacy violated?
  - This is a problem *even if Alice was not in the database!*
  - This is exactly the sort of information we want to be able to learn…

# What is Privacy?

Privacy Definition, Attempt 2:

*An analysis of a dataset D is private if the data analyst knows almost no more about Alice after the analysis than he would have known had he conducted the same analysis on an identical database with Alice's data removed.*

# Differential Privacy [Dwork-McSherry-Nissim-Smith 06]



# Differential Privacy

$X$: The data *universe*.

$D \subset X$: The dataset (one element per person)

Definition: Two datasets $D, D' \subset X$ are *neighbors* if they differ in the data of a single individual. i.e. $|D \,\Delta\, D'| \leq 1$.

# Differential Privacy

$X$: The data *universe*.

$D \subset X$: The dataset (one element per person)

Definition: A mechanism $M: 2^X \rightarrow R$ is $(\epsilon, \delta)$-differentially private if for all pairs of neighboring databases $D, D' \subset X$, and for all events $S \subseteq R$:

$$\Pr[M(D) \in S] \leq e^{\epsilon} \Pr[M(D') \in S] + \delta$$

$(1 + \epsilon)$

Definition: A mechanism $M: 2^X \rightarrow R$ is $(\epsilon, \delta)$-differentially private if for all pairs of neighboring databases $D, D' \subset X$, and for all events $S \subseteq R$:

$$\Pr[M(D) \in S] \leq e^{\epsilon} \Pr[M(D') \in S] + \delta$$

- Think of $\delta$ as exponentially small (or even 0)
- Think of $\epsilon$ as a small constant.
  - If $M: 2^X \rightarrow R$ is $(\epsilon, 0)$-DP, and $|D \Delta D'| = k$, then: $\Pr[M(D) \in S] \leq e^{\epsilon k} \Pr[M(D') \in S]$
    - So nothing useful is possible for $\epsilon = o(\frac{1}{n})$

# Why is Differential Privacy "Privacy"?

- It should guarantee "freedom from harm"
- A useful fact – resilience to post-processing:
  - For any $f: R \to R'$, and any $(\epsilon, \delta)$-differentially private $M: 2^X \to R$, $f \circ M: 2^X \to R'$ is also $(\epsilon, \delta)$-differentially private.
- What if $f$ maps mechanism output to events you care about?
  - Differential privacy: "Except for rare events that occur with probability $\leq \delta$, your future utility will decrease by at most a $(1 - \epsilon)$ factor by participating in the database."

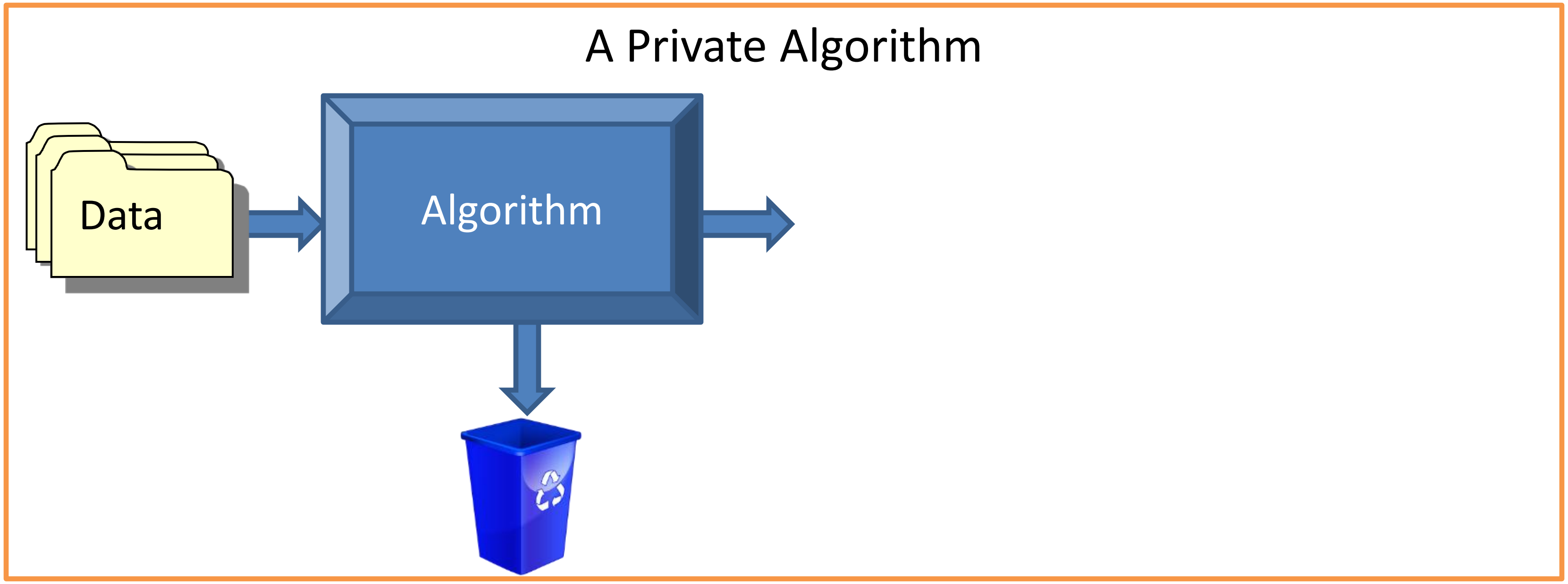
# Why is Differential Privacy “Privacy”?

- $f$ incorporates any auxiliary information an analyst may have about the database now *or in the future*.
- The guarantee is just as strong *even if the analyst knows the entire database except for your value*.
  - A worst case model: no longer any need to reason about what the analyst knows.

# So now we have a definition.

## Course Roadmap

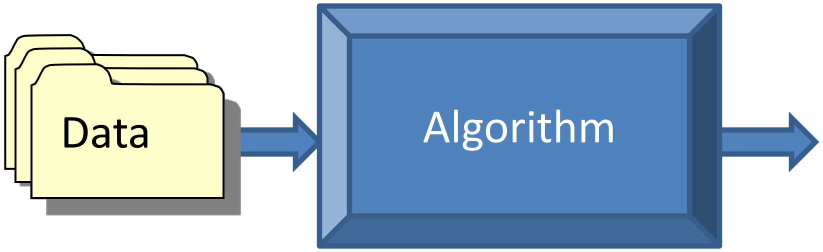
- What are the big questions?
  - How do we trade off privacy and utility?



# So now we have a definition.
## Course Roadmap

- What are the big questions?
  - How do we trade off privacy and utility?

A Useful Algorithm



# So now we have a definition.
## Course Roadmap

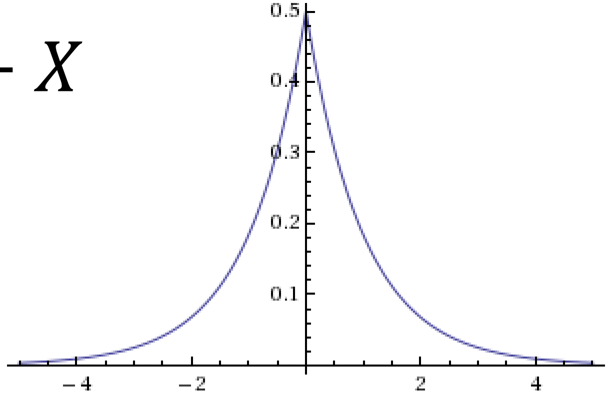
- How can we build useful, differentially private algorithms?
  - Out of basic building blocks, glued together by composition theorems.

# So now we have a definition.
## Course Roadmap

- Basic Building Blocks
  - Answering numeric queries through perturbation

$$M_f(D) = f(D) + X$$

$$X \sim Lap(\frac{1}{\epsilon})$$



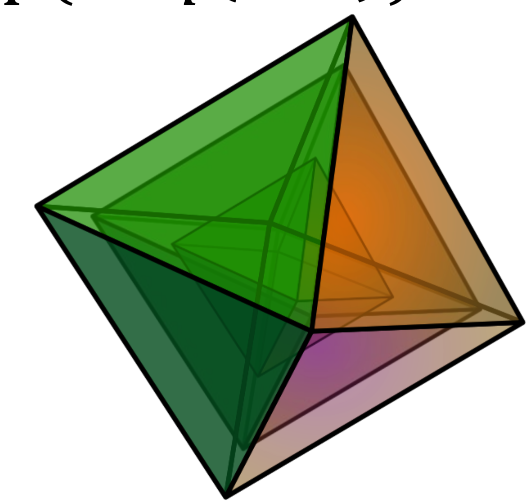
# So now we have a definition.

## Course Roadmap

- Basic Building Blocks
  - Answering non-numeric queries by sampling from a private distribution

$$M_q(D): 2^X \to R$$

Output $r \in R$ with probability $\sim \exp(-\epsilon q(r, D))$



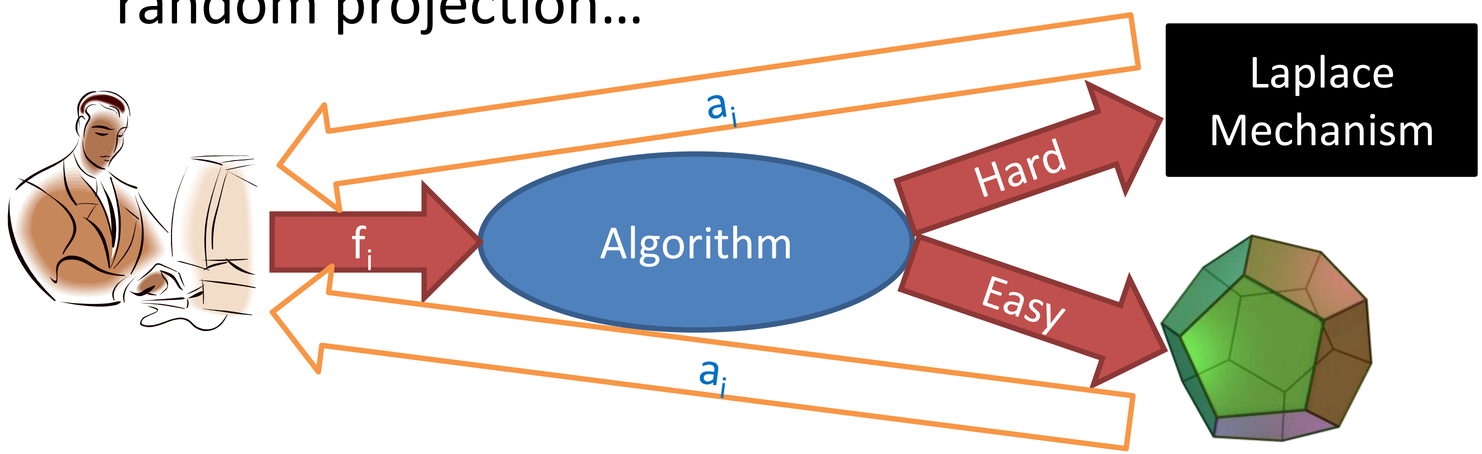
# So now we have a definition.
## Course Roadmap

- Combining building blocks into algorithms
  - What are the privacy guarantees for an algorithm $M$ composed of $k$ subroutines $A_1, \dots, A_k$ that are each $(\epsilon, \delta)$-differentially private?
    - $(k\epsilon, k\delta)$-differentially private
    - *Also* $\approx (\sqrt{k \log \frac{1}{\delta'}}\epsilon, k\delta + \delta')$-differentially private
      - Can *trade* lots of $\epsilon$ for a little more $\delta$.

# So now we have a definition.

## Course Roadmap

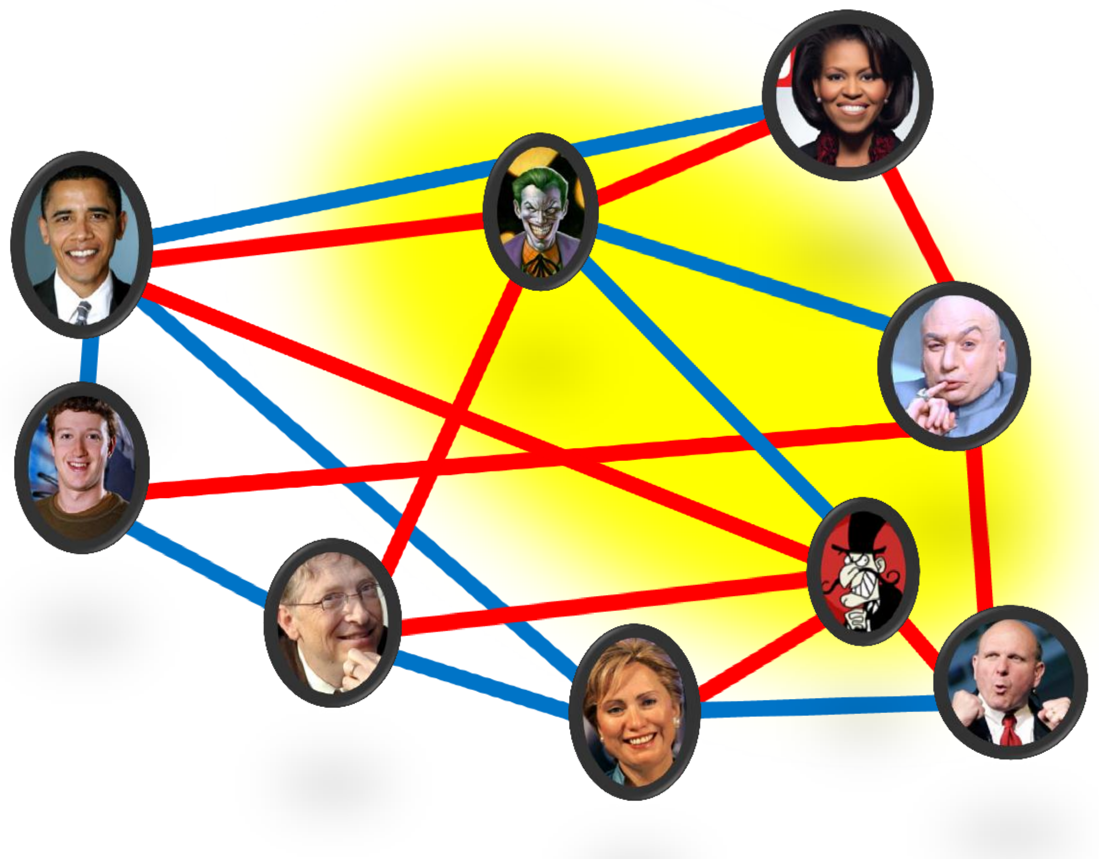
- What can we build?
  - Algorithms for accurately answering *exponentially* many numeric queries in the database size!
    - Leveraging machine learning theory, compression, random projection…



# So now we have a definition.

## Course Roadmap

- What can we build?
  - Algorithms for combinatorial optimization

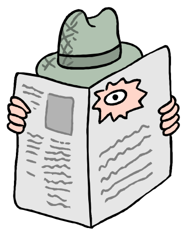


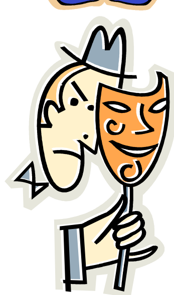
# So now we have a definition.
## Course Roadmap

- What can we build?
  - Streaming Algorithms
    - That are private even if a hacker is able to look at the internal state of the algorithm.

# So now we have a definition.
## Course Roadmap

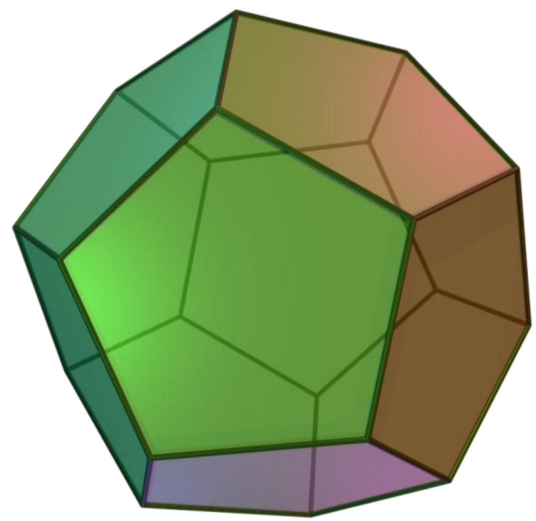
- What can we build?
  - Auctions and truthful mechanisms for privacy-aware economic agents






# So now we have a definition.
## Course Roadmap

- What *can't* we build?
  – Lower bounds from linear programming
    - Answering queries *too* accurately lets an adversary reconstruct the database



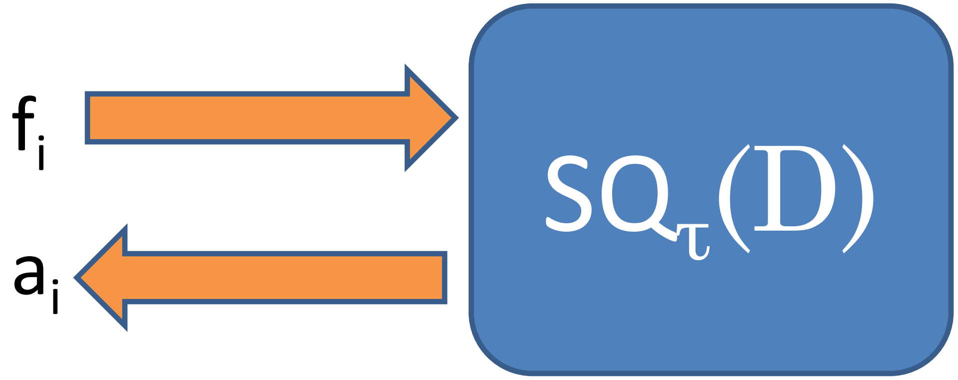
# So now we have a definition.
## Course Roadmap

- What *can't* we build?
  - Lower bounds from packing arguments
    - The existence of good *error correcting codes* give lower bounds in differential privacy

# So now we have a definition.

## Course Roadmap

- What *can't* we build?
  - Lower bounds from learning theory
    - Efficient query release algorithms in Kearns' *statistical query* model would lead to too-good-to-be-true learning algorithms.



# To Muse On:

- Think about why differential privacy protects against blatant non-privacy
- Read [Narayanan,Shmatikov06]: How to de-anonymize the Netflix data set.